1. What is the output of the following code:

```java
int s = 0;
int t = 1;
for ( int i = 0 ; i < 10 ; i++ ) {
    s = s + i;
    for ( int j = i ; j > 0 ; j-- ) {
        t = t * (j - i);
    }
    s = s * t;
    System.out.println("t = " + t);
}
System.out.println("s = " + s);
```

```
t = 1
t = 0
t = 0
t = 0
t = 0
t = 0
t = 0
t = 0
t = 0
s = 0
```

2. Given an array of String. Each element in the array contains a String object. Example:

```
“Sameh” “Eliot” “Sarah” “Dalia” “Nidal”
```

Write a method with the following signature:

```java
public String combine(String[] strings)
```

Which combines the Strings to form one String that would contain the first letters of each String, followed by the second letter from each string, etc.... You may assume that all Strings have the same length, but you may not assume that the array only has 5 entries. The above array would return the following String:

“SESDNalaaimirldeoiaiahthal”

```java
public class Question2 {

    public static String combine(String[] strings) {
        // define the return variable
        String s = new String();

        // find the length of all the Strings by looking at the
        // length of the first String
```
```java
int strLength = strings[0].length();

// i represents the character position within each word
for (int i = 0; i < strLength; i++) {
    // j represents which word within the array
    for (int j = 0; j < strings.length; j++) {
        // get the i'th character from the j'th word
        s = s.concat(strings[j].substring(i, i+1));
    }
}
// return the combined String
return (s);
}

public static void main(String[] args) {
    String[] strings = new String[5];
    strings[0] = new String("Sameh");
    strings[1] = new String("Eliot");
    strings[2] = new String("Sarah");
    strings[4] = new String("Nidal");
    System.out.println(combine(strings));
}
```
3. Given the following classes, show the output from running the Test class and give an explanation:

```java
public class Vehicle {
    public Vehicle() {
        System.out.println("New Vehicle");
    }
    public void drive() {
        System.out.println("Vehicle: drive");
    }
}

public class Car extends Vehicle {
    public Car() {
        System.out.println("New Car");
    }
    public void drive() {
        System.out.println("Car: drive");
    }
}

public class SportsCar extends Car {
    public SportsCar() {
        System.out.println("New SportsCar");
    }
    public void drive() {
        System.out.println("Sports Car: drive fast");
    }
}

public class Test {
    public static void main(String args[]) {
        Vehicle v = new Vehicle();
        Car c = new Car();
        SportsCar sc = new SportsCar();
        v.drive();
        c.drive();
        sc.drive();
        v = c;
        v.drive();
        c = sc;
        c.drive();
        v = sc;
        v.drive();
    }
}
```

New Vehicle
New Vehicle
New Car
New Vehicle
New Car
New SportsCar
Vehicle: drive
Car: drive
Sports Car: drive fast
Car: drive
Sports Car: drive fast
Sports Car: drive fast

4. Write a for loop to compute the sum \( 1^2 + 2^2 + 3^2 + 4^2 + 5^2 + \ldots + n^2 \). Assume that \( n \) is a variable that has already been defined.

```java
for (int i = 1; i <= n; i++) {
    sum += i*i;
    System.out.println("for " + i + " the sum = " + sum);
}
```
5. Show the output from running the Test class:

```java
public class Test {
    public static void main(String args[]) {
        Integer i1 = new Integer(9);
        Integer i2 = new Integer(9);

        if (i1 == i2)
            System.out.println("Test1");
        if (i1.equals(i2))
            System.out.println("Test2");
        if (i1.equals(new Integer(9)))
            System.out.println("Test3");
    }
}
```

Test2
Test3

6. Write a complete Java class named MyClass that has the following private attributes:

   a. myInts, an array of int's that has a maximum capacity of 100.
   b. numInts, an int variable that keeps track of the number of elements in myInts.

And the following methods:

   a. public MyClass() – Constructor that initializes myInts and numInts.
   b. public int addInt(int i) – Adds i to myInts and updates numInts. Returns the index where i was added. If there is no room to add i to myInts, return -1.
   c. public int find(int i) – Find the first occurrence of i in myInts and return its index, return -1 if not found.
   d. public int getInt(int i) – Return the integer at index i if it exists, return -9999 otherwise.
   e. public boolean isFull() – Returns true if myInts is full, false otherwise.
   f. public boolean isEmpty() – Returns true if myInts is empty, false otherwise.

```java
public class Question6 {
    private int[] myInts;
    private int numInts;

    public Question6() {
        myInts = new int[100];
        numInts = 0;
    }

    public int addInt(int i) {
        int index = -1;
        if (!isFull()) {
            index = numInts;
            myInts[index] = i;
            numInts++;
        }
    }
}
```
public int find(int i) {
    int index = -1;
    if (!isEmpty()) {
        for (int j = 0; j < numInts; j++) {
            if (myInts[j] == i) {
                index = j;
                break;
            }
        }
    }
    return index;
}

public int getInt(int i) {
    int num = -9999;
    if (!isEmpty() && (i <= numInts - 1)) {
        num = myInts[i];
    }
    return num;
}

public boolean isFull() {
    boolean answer = true;
    if (numInts < myInts.length) {
        answer = false;
    }
    return answer;
}

public boolean isEmpty() {
    boolean answer = false;
    if (numInts == 0) {
        answer = true;
    }
    return answer;
}

public static void main(String args[]) {
    Question6 q6 = new Question6();
    System.out.println(q6.isEmpty());
    System.out.println(q6.isFull());
    for (int i = 1; i <= 100; i++) {
        System.out.println(i + " Added at index " + q6.addInt(i));
    }
    System.out.println("150 was found at index " + q6.find(150));
System.out.println("int at index 55 is " + q6.getInt(55));
System.out.println("int at index 80 is " + q6.getInt(80));
System.out.println(q6.isFull());
}